

A Player-like Agent Reinforcement Learning Method For Automatic Evaluation of Game Map

Game map is an important human-computer interactive content-bearing platform in major games. With the application of cellular automata(CA) and Procedural Content Generation (PCG)in map generation, the spatial scale and data volume of current game maps are increasing greatly, while in game map test procedure, automatic methods such as interactive test script are inadequate both in depth and application breadth, especially in the lack of game map evaluation from player experience perspective. This research proposes an automatic game map test method based on agent reinforcement learning. By establishing agents' interactive action models standing for different types of players' behaviors in the map, universal evaluation of the map environment is enhanced through agent actions, which can optimize game map design from the perspective of player experience with quantitative value of inferiority. Finally, our campus scenes in Minecraft were used as the experimental environments to verify the effectiveness of the method.

CCS CONCEPTS • Games and Play • Computational Interaction

Additional Keywords and Phrases: Games/Play ; Machine Learning ; Programming/Development Support ; Artifact or System ; Method ; Theory ; Application Instrumentation / Usage Logs ; Quantitative Methods ; Usability Study

ACM Reference Format:

[1] INTRODUCTION

This paper introduce an modified reinforcement learning model to solve the problem of evaluating game maps from the perspective of player experience with player-like agents. As the demand for qualified ever larger game map than before in immense games rises,the research of game map testing, especially the automatic testing from the perspective of player experience, now deserves additional attention.

1) In recent years, the magnitude and complexity of modern game maps are exploding with the assistance of PCG(Procedural Content Generation) methods. For example, Assassin's creed, a AAA game, is constantly updated over the past 10 years with the exponential growth of game maps as 1700 times from 0.13 square kilometers in Damascus to 230 square kilometers in the North Sea area of Europe,

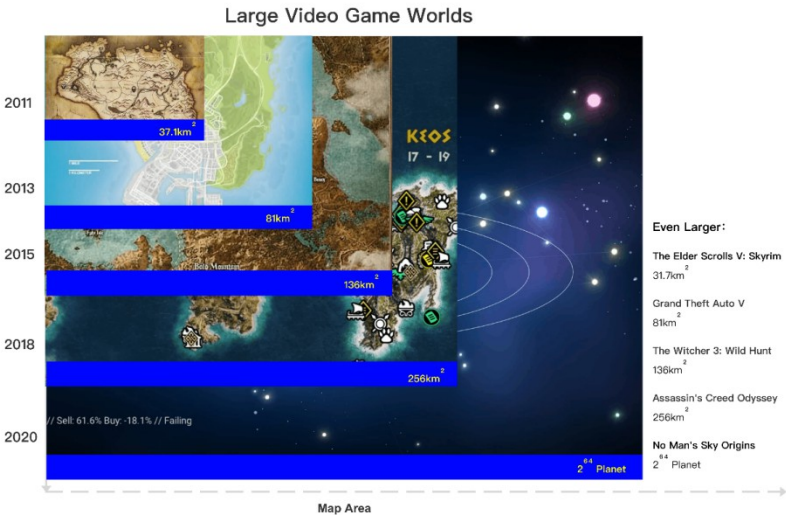


Figure 1: Evolution of game map

Automatic procedural modeling of game map (mainly PCG) has always been an academic research frontier for over thirty years, resulting in high-quality procedures for specific game map features of any types[1], such as landscapes[2, 3], rivers[4-6], plant models[7] and vegetation distribution [8], road networks[9], urban environments[10], and building facades[11-13]. [1] introduced declarative modeling of virtual worlds that combines the integrated use of various procedural modeling techniques with a semantics-driven model to capture designer's intent. [14] enables the construction of a 3- dimension buildings using the grown building footprints by L-system for amateur players to create custom game content. These PCG achievements have formed engineering application tools such as Houdini, which can produce massive and large-scale game maps quickly.

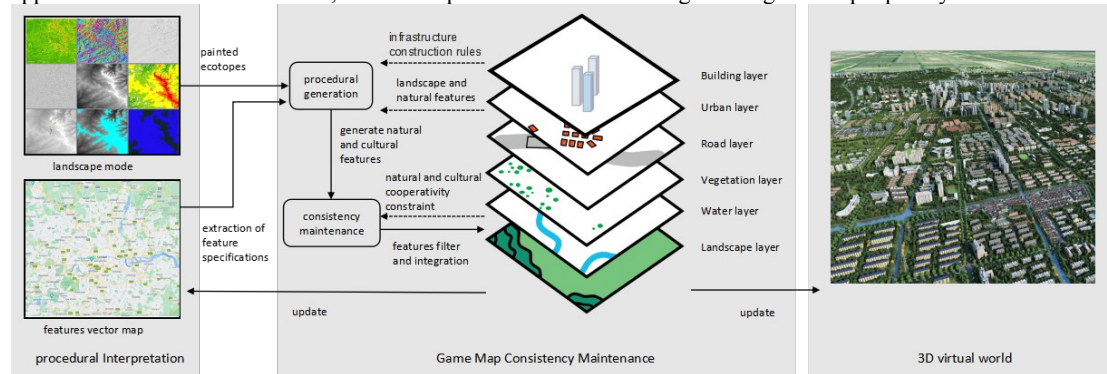


Figure. 1 An overview of the workflow of procedural game map generation and objective consistency maintenance[15-17]

And many studies emerged on ensuring the consistency and compatibility of various elements of the PCG map objectively. [18] proposed a shader-based system for real-time integration of Geographic Information Systems (GIS) vector features, such as road and rivers, into a DEM. [19] presented an interactive simulation system for cities growing over time, by expanding streets in the city's road network. A dynamic system that connects geometrical with behavioral modeling is also proposed. [20] applied evolutionary and other metaheuristic search algorithms to automatically generating content for games, both digital and nondigital (such as board games). Despite [15] discussed the consistency of all generated content of various procedural models that it goes far beyond the internals of individual procedural methods. These objective verification methods (also called static test), which are often implemented by automated test scripts, are testing generated game map according to some computable criteria (e.g., Is there a path between the entrance and exit of the dungeon[24], or does the tree have proportions within a certain range? or a fully automated process using image processing techniques to compare and judge examples[21]). If the test fails, all or some of the candidate game map is discarded and regenerated, and this process continues until the content is good enough[20].

2) While subjective game map testing is failed to match the level of automatic generation of game map.

The subjective evaluation (also called dynamic test) for game map may be a human observer who specifies which individual players survive in each map generation[22, 23]. Two traditional types of manual subjective testing methods to game map: public testing and internal testing hindered PCG applications in game industry. Public testing has high efficiency in testing the scope of game content (not only including maps), but it needs high manpower by advertisement and other costs to promote the participation of the public. Internal testing by the internal personnel of the game company is not enough to cover a big map range. On the other hand, internal testers cannot evaluate the experience of map design schemes on behalf of public players. This is also the reason for the public beta of MMO games in recent years, such as *World of Warcraft* and *JX Online 3*.

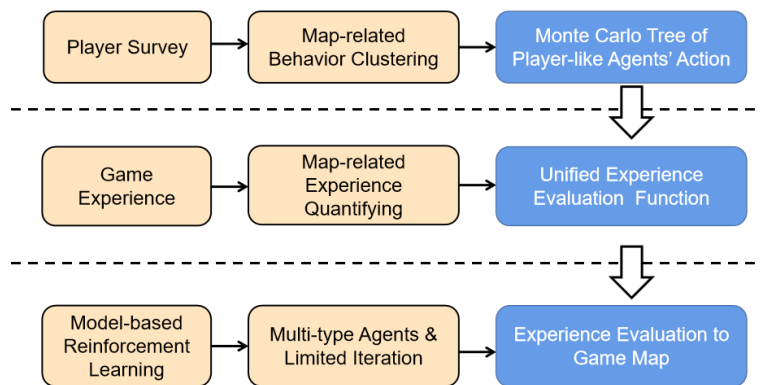
3) Two possible solutions do not solve the problem of subjective test of game map so far.

One possible solution is to involve human players in the PCG rather than in the testing process afterward. In PCG DESIGN METAPHORS, The PLAYER EXPERT[25] is supposed to encompass any analysis, interpretation, and adaptation suggestions specifically related to player experience in any use of PCG that uses player behaviour and experience as input. Kazmi and Palmer[26] describe a system, embodying both a PLAYER EXPERT and a DESIGNER, premised on analysing and interpreting player actions in terms of player skill and style. [27] proposed an interactive process between the player and the computer which allows the player to guide evolving equations by observing results and providing aesthetic information at each step of the procedural models and achieving flexible

complexity. This kind of solution slows down the entire map generation process and requires player EXPERTs having considerable knowledge of PCG. Moreover, a few player EXPERTs involved in the PCG cannot represent all public players and cannot validate game maps by themselves, expensive public testing are still the most reliable game map testing methods so far.

The other possible solution is to make objective automatic tests more subjective that is to empower automated test scripts with prior knowledge so that these artificial players [28–30] have evaluation abilities closer to the human player's experience of game map than before without efficiency losing or cost increasing. Game-playing agents are beneficial in play testing by reducing costs and the need for human play testers [31]. Such as MCTS and reinforcement learning (RL) models can present automated play testing methods without the need of human player intervention. AI agents have been found useful in finding bugs [32], game parameter tuning [33]. The results in behaviors of RL agents more closely resembling those of human players than traditional Objective verification methods, thus increasing the probability of finding bugs and exploits. Recent techniques have tackled these kinds of scenarios using either a single model learning the dynamics of the whole game [34], or two models focusing on specific domains respectively (navigation and combat) [35]. Devlin et al. showed how observations of human play data can be used to bias MCTS to play the card game Spades [36]. They use a relative entropy measure to assess the similarity of playing styles to traces of human players. Zook et al. limited the computational resources of MCTS to simulate player skill for a number of games [37] and similar findings were reported by Nelson [38]. Another approach to biasing the MCTS search process to be more similar to human players is described by Khalifa et al. [39]. Christoffer Holmgard et al. [40] bias the MCTS using evolution applying designer-defined utility to produce a set of personas that show what different playstyles might look like in MiniDungeons 2. [41] introduced a self-learning mechanism to the FPS type game testing, in which the required sum of game frames to reach a certain percentage of max reward (the agent well-trained) are regard as an quantitative indicator of the difficulty of the game environments. The shortcoming of previous studies is the behaviors of these agents are not directly adhere to the behavior of real players, but reinforced by reward guidance under different navigation targets. These navigation targets are not as same as the goals of players' interaction in the game map, and the training environments or methods are simplified in different degrees compared with the real games.

Framework of Game Map Centered Playtesting



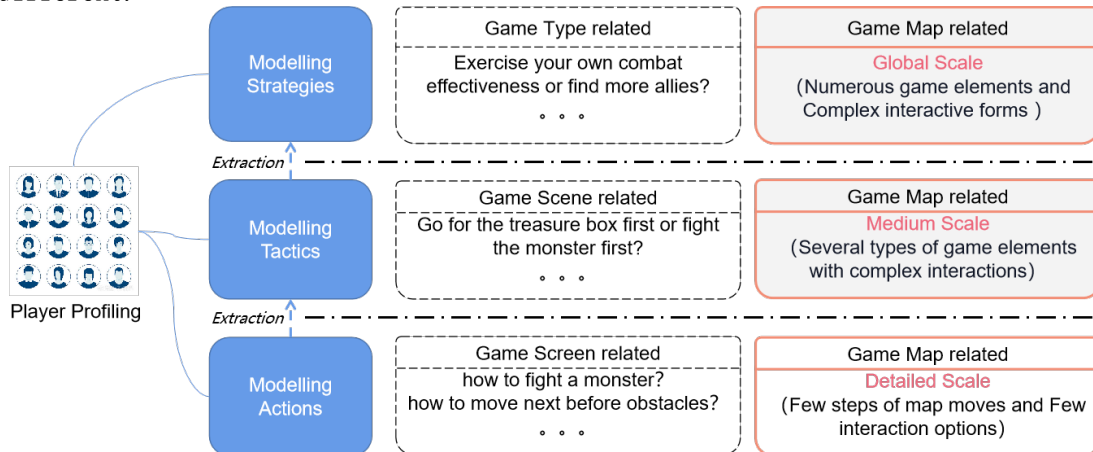
Therefore, our work advances the state of automatic game map testing in model-based reinforcement learning with player-like agents. The workflow of this research is shown as figure:

- 1) Game Player Behavior analysis and Clustering for game map testing.
- 2) Constructing Player-like Experience Evaluation Model of Game Map.
- 3) Modifying a Model-based reinforcement learning for game map evaluation.
- 4) Experiments in Minecraft map testing

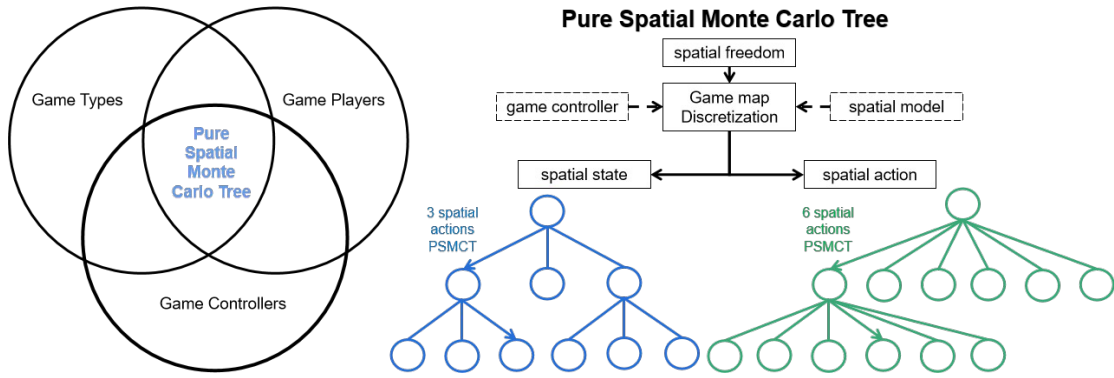
[2] GAME PLAYER BEHAVIOR CLUSTERING FOR GAME MAP TESTING (康崇卓)

In our research, player behavior model focusing on the automated test of game map exclusively, and then to establish a map related behavior MCT (Monte Carlo Tree) model which can drive an artificial agent as a policy function. Human player behaviors in any games can be regarded as a sequential decision-making, A Markov decision process (MDP) represents a formal framework to describe such a process. Modeling the possible interaction between an arbitrary

agent and its environment over time. The MDP method requires the human player behavior model to accurately define various states and various direct actions. Generally, Monte Carlo Tree Search (MCTS) is an alternative method to solve MDPs. It estimates the optimal action by building a tree of possible future (game) states and rewards, with each tree node corresponding to the state resulting from an explored action[29]. Obviously, for different human players and different game types, the structure of Monte Carlo tree could be very different.



Map related player behaviors has two characteristics[40], 1) widely existed; 2) elementary interactions. First.[42]According to Bakkes, player behavior modeling can be divided into four levels, player analysis, strategic level, tactical level, action level. This behavior system migrated from the field of military command can be found in various games, as shown in the above figure. Among the four different behavioral levels of various games, map interaction is indispensable. Secondly, the basic element characteristics of map interaction behavior are also obvious. As a typical discrete space area, game map can support a limited number of player behavior, including spatial dimension switching, speed switching and switching frequency change.[43]. Pure map interaction elements are simple and identical in the analysis of game interaction in various studies[44], Discrete definitions of degree of freedom and moving distance in map space, such as upper and lower left and right movement and moving step size[45, 46], Further define spatial intersection, spatial aggregation, etc., to generate interactive behavior with other game interaction elements (shooting target, treasure box). [43]The spatial position, movement speed and current direction of the avatar are the game map related player behaviors without any other social properties such as level, health, strength or any attractions in game scene such as rewards items, flags, monsters which diversified in different types of games. Among them, [44]Movement through virtual worlds is one of the primary mechanics in open-world (sandbox) games.in other words, and MoveDistance is a player behavior highly related to the desire of game exploration[47]. [48]proposed Landmarks are usually used by players for pathfinding. Each type of players having specific moving pattern (spatial decision tree) about transition probabilities between landmarks.



Thus, we propose Pure Spatial Monte Carlo tree (PSMCT) as the basic framework of the map test behavior model, as shown in the following figure. The basic map interaction elements are closer to the game character's own space roaming capabilities, and are also more in line with the player interaction behavior model purely for game map automatic testing purposes. PSMCT only contains the definition of the basic interactive elements and basic states of the game map.

[3] PLAYER-LIKE EXPERIENCE EVALUATION MODEL OF GAME MAP

From the perspective of human-computer interaction, the game experience of game players is a very personalized and comprehensive concept, which contains rich elements from arousal of endogenous emotion [49] to engagement with external player game duration and game frequency expression [29]. Most of previous researches rely upon the assumption that player emotions can be inferred via the association of player self-reports (Subjective player experience modeling, SPED) and game context variables (objective player experience modeling, OPEM) [50, 51]. But there is usually significant experimental noise in SPED which may be caused by player learning and self-deception effects. Either, self reports in SPED can be intrusive if questionnaire items are injected during the gameplay sessions While afterward questionnaire items are suffering from minimal post-experience effects [52–54], The objective PEM approach can be model-based or model-free. Model-based refers to emotional models derived from emotion theories (e.g., cognitive appraisal theory [55], usability theory [56], belief-desire-intention model, the cognitive theory by Ortony, Clore, & Collins [57], Skinner's model [58], Scherer's theory [59]), but there are also theories about player affect that are specific to games, such as Malone's design components for fun games [60], Koster's theory of fun [61], and game-specific interpretations of Csikszentmihalyi's concept of Flow [62]) such as the popular emotional dimensions of arousal and valence [63], [64] Model-free PEM refers to the construction of an unknown mapping (model) between modalities of player input and an emotional state representation via player annotated data [65], The key limitations of the OPEM approach include its high intrusiveness, low practicality (specific to games combined with high complexity), and questionable feasibility.

Our study only selects the game map exploration within a limited time span as the objective evaluation indicator of game map experience. The reason is that game map exploration is the main basis of high-level game experience. Although the game experience metrics of OPEM and SPED are quite different, the level of experience is recognized. Spatio-temporal features of game interaction (in our study, PCMCT) are usually mapped to levels of cognitive states such as

attention, challenge, and engagement [51].and the player' s cognitive processing patterns and cognitive focus may influence emotions (affective states: fun, challenge, frustration, predictability, anxiety, and boredom[66]). Ferro etc. [67]proposed Game Experience and Elements (GEM) framework . Through exploratory factor analysis (EFA) method, it is also determined that game map exploration is the basis of game experience and the most important cognitive element.

In order to calculate and cooperate with the test agents of game map by traversal of PCMCT conveniently, this study proposes an exploration based game map experience function (EBGMEF). The calculation formula of the exploration degree of the game map is based on three assumptions:A. The game map is spatially uniformly discretized, such as a uniform hexagonal grid (as in Civilization 6, Total War etc.), a uniform quadrilateral or cube (Flame Heraldic Series, Minecraft). This assumption can decompose the overall experience value of the map into the sum of the experience values of each uniform discrete unit. B. game exploration is time-relative, influenced by the player 's total game time. In previous researches about the degree of player involvement or fatigue,the important indicators like frequency of operation is also counted within a specified time[68, 69].In game map test, utilizing a limited play time (may be defined by how long human player plays once at average) to count the size of the explored map scope is clear and cohering to human player' s feel. C. Game exploration is related to players ' memory. The experience of game map varies with players' [70, 71]ability of spatial memory , and this spatial memory is the remaining value of players ' map-seeking, especially the instant impressions of foggy games (such as Star Wars, age of empires) after map exploration. Obviously considering the spatial memory ability of different players can better illustrate the experience value of game maps for different players than without it.

In the above formula of EBGMEF, a game map consists of n discrete units, each unit have an initial experience value of 1. When the player 's agent roams to the current map unit for the first time, the quality value of the current map unit is calculated only once(that means exploration). k is the maximum number of memorized map units of a certain type of player, j indicates that the j th map unit on this memorized path, such as 10 map units can be remembered by a certain play-like agent, k is 10, j is in $\{ 0 \sim 9 \}$, and k is determined by map memory rate Y and a memory threshold which does not appear in the above equation.A memory threshold are used to eliminate map units with little impression, for example if . memory threshold is set to be 0.01,and Y is 0.8,then after 20 map units,the remaining memory of the former 21 map unit is less than 0.01,then, the maximum remembered map units k is 20.Obviously, the farther the map unit is from the current unit ($i = 0$), the less the spatial memory value is left(short-term memory characteristic of human beings [71]) .

Two points worth noting about the formula that,firstly the experience value of the discrete unit of the game map is non-renewable, and the experience calculation will not be carried out when the player-like agent passes again, which also conforms to the common sense of exploration as one time discovery, and the more frequent the player-like agent passes, the more boring the design of the game map is[69, 72]. Secondly

The total experience value of game map is positively correlative to the total time of agent exploration. Various Spatial Traversal Algorithms Based on Greedy Algorithm[24] can explore a map completely as long as there is enough time, it is obvious that the efficiency of time-related exploration of the map can better reflect the **player-like** experience of a game map.

[4] MODEL-BASED REINFORCEMENT LEARNING FOR GAME MAP EVALUATION

Model-free reinforcement learning (RL) can be used to learn effective policies for complex tasks with basic interactions between agents and the environment with reward rules for agents, such as AI played Atari games[73] from image observations of the running games. However, this typically requires very large amounts of interaction data, also takes a long computing process for agents to learn, such as [74] OpenAI 5 used about equivalent 10000 years of human 's game time to outperform the human world champions at an esports game Dota2.

Model-based reinforcement learning (RL) can use the known behavior or environment models to set the action policies of the agents, conduct automatic learning in specific types of data enhancement, or shape the hidden space in the time domain with substantially improved efficiency by applying predefined models[75]. Using models of environments, or informally giving the agent ability to predict its future, has a fundamental appeal for reinforcement learning[75, 76]. The spectrum of possible applications is vast, including learning policies from the model[77-82], capturing important details of the scene [83], encouraging exploration[84], creating intrinsic motivation[85] or counterfactual reasoning[86].



Therefore, we propose a Model-based reinforcement learning based on PCMCT and EBGMEF, this reinforcement learning in this study is different from previous game reinforcement learning models. The uniqueness includes :

- 1) Action strategy player-like : The agent i ' s action strategy function comes from the fixed action strategy model (PCMCT) of a specific type i of players. The agent does not need training to improve the action strategy, and that ensuring the agent ' s behavior close to the human player.
- 2) Experience reward player-like: the reward R_i obtained by agent i ' s roaming action through EBGMEF reflects the human player 's experience of game map as the exploration memory, rather than as the stimulant for training agent ' s behavior.
- 3) Spatial memory player-like : The value of map unit Q_i comes from the direct action reward R_i of agent i with the spatial memory rate. If the Player i more

proficient in playing game, the higher the spatial memory rate is [69, 72], this configuration ensures the experience assessment close to human player.

4) Map total evaluation player-like : According to the above RL evaluation model of this study, the total value of identical game map varies with the agent types, and the total values of different game maps differentiate for the same agent type i , that is cohering to human players' testing.

[5] EXPERIMENTS

This study uses Minecraft as the test environment. First of all, Minecraft is popular in game research community with great potential for automated world map testing [87]. There are a large number of Minecraft maps shared on the network, including Hogwarts School of Magic, King's Landing of the Song of Ice and Fire, Berkeley University of California, Beijing University of Posts and Telecommunications and so on. It provides almost unlimited game map test resources. Secondly, Minecraft map automatic test calculation is simple. Minecraft map is a standard octree discrete space with uniform unit size [88], The roaming actions in Minecraft of the player is clear and the state-action calculation is simple; third, the total development workload of Minecraft map automatic iterative test is low. Microsoft has published Malmo reinforcement learning environment and open source its code, on which we modified to players-like reinforcement learning model in this study.

The version of Malmo used in this study is 0.37.0, the program language used for rewriting is JAVA, the modification includes 3 steps:

A. Extend the map base class of Malmo. First, the map unit has the initial exploration value (represented by the red rose on the map block) as a default attribute, and then each map unit saves its own experience value only once during an agent test when the agent first passes the map unit;

B. The map test agent is built with a inside PCMCT. Firstly, the PCMCT of the agent is consistent with the clustering results of the player survey in our experiment, representing the roaming behavior of a certain type of player. Secondly, the agent is responsible for maintaining the map units memory queue. The queue length depends on the memory rate and the forgetting threshold, for example if the memory rate is 0.8 and the forgetting threshold is 0.01. The map units after 20 map cells does not meet the forgetting threshold ($0.0821001 < 0.01$), and the memory queue of the agent is set to 20.

C. Global test configuration is added. Includes current test map files, number of test agents, and test time.

Besides the modification of Malmo, the emphasis of the experiment is to establish the player-like PCMCT. In this study, the transfer probability of map state and agent action in player PSMCT modeling is obtained by questionnaire, for player survey is more operable and universal in map testing task than others so far. Sharma et al. [89] proposed a higher-order classification of player modelling, in which a distinction is made between (1) direct-measurement approaches (e.g., that utilise biometric data) and (2) indirect-measurement approaches (e.g., that infer the player's skill level from in-game observations). [90] analyzing game log data shows that experienced players often try more spatial choices in games. [91] In a specific type of game scenario, the potential field of the game scenario is established through multiple statistics of player behavior, and then the AI agent is driven by the potential field gravity in different regions. Obviously, although multidimensional clustering

and other methods[92]can effectively deal with the game behavior log data, the contents of log data of different game types are very different. For example, the location of a treasure box or a monster, which is not necessarily existing in a PCG game map, will affect the player 's behavior. In summary, for game map test, it is difficult to guarantee the representativeness and universality of state-action learning by behavior log data acquisition or in game observation of any specific game.

Therefore, we invited human players to answer the questionnaire as a Delphi method. The design of the questionnaire includes two types of questions, one category is the classification questions on player experience[93] ; the other is map state-action questions. In 2015 Rafet Sifa et al. [94] found that players' game time determining the player behaviors as the dominant feature, through statistics of a large number of players' data on the steam platform. But due to the differences caused by game types, their research does not involve the roaming behavior clustering in game maps. In the study of StABLE player behavior model proposed by Fragoso et al. [44], the advanced players and non-advanced players divided by game experiences show differentiation in playing behaviors (interaction frequency, moving distance, etc.), and this differentiation has high stability in all scenarios. Referring to the above researches, our player experience classification problem takes the total game duration, number of games played and frequency of playing games as the criteria for player classification.

Referring to the above PSMCT, our Minecraft game map state-action possibility assessment problems are processed in 3 steps, first is to define a variety of map states represented by representative landmarks in Minecraft , then investigate the player ' s state-action selection and action range (controls of agent speed). Finally, according to the answers, clustering the Minecraft map behavior data by different types of players and set up state-action functions of PSMCT by sampling probability.

Table 1 Main survey elements

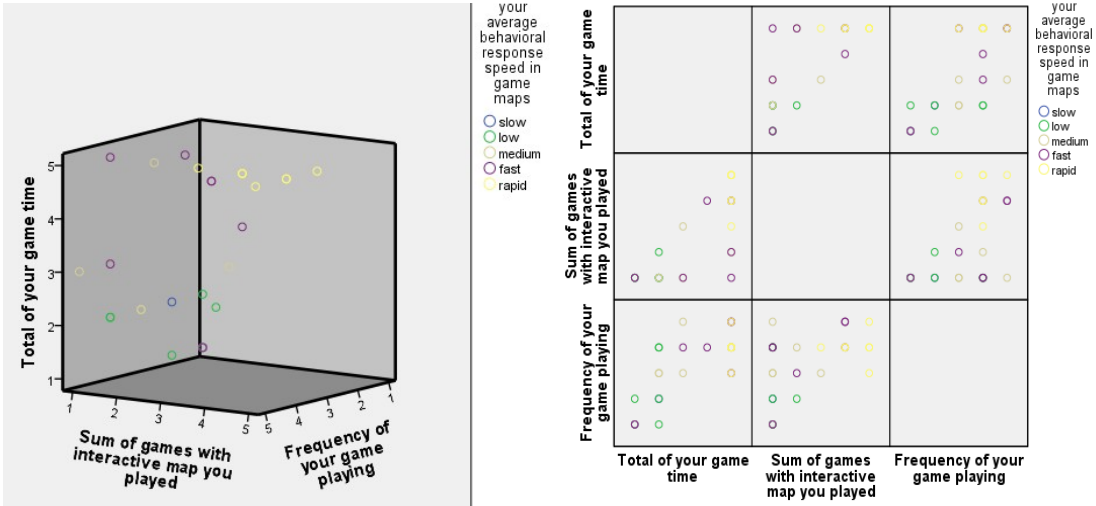
Player classified variables	Questionnaire items
Total Time of player Games	How many hours to play
Number of game players involved	How many kinds of games have played (limited to any game with interactive map)
player game frequency	average time interval for game play
Game map action policy	Questionnaire items
Map State	Front Obstacles, One Side L Obstacles, Two Sides U Obstacles, Front Step Up, Front Step Down, Wide Pavement, Three-Way Road
Map Action	Forward, Backward, Right, Left and Jump
Map action range	very large, large, medium, small, tiny
Map Action Switching Frequency	Rapid, Fast, Medium, Low, Slow
Map memory rate	very large, large, medium, small, tiny

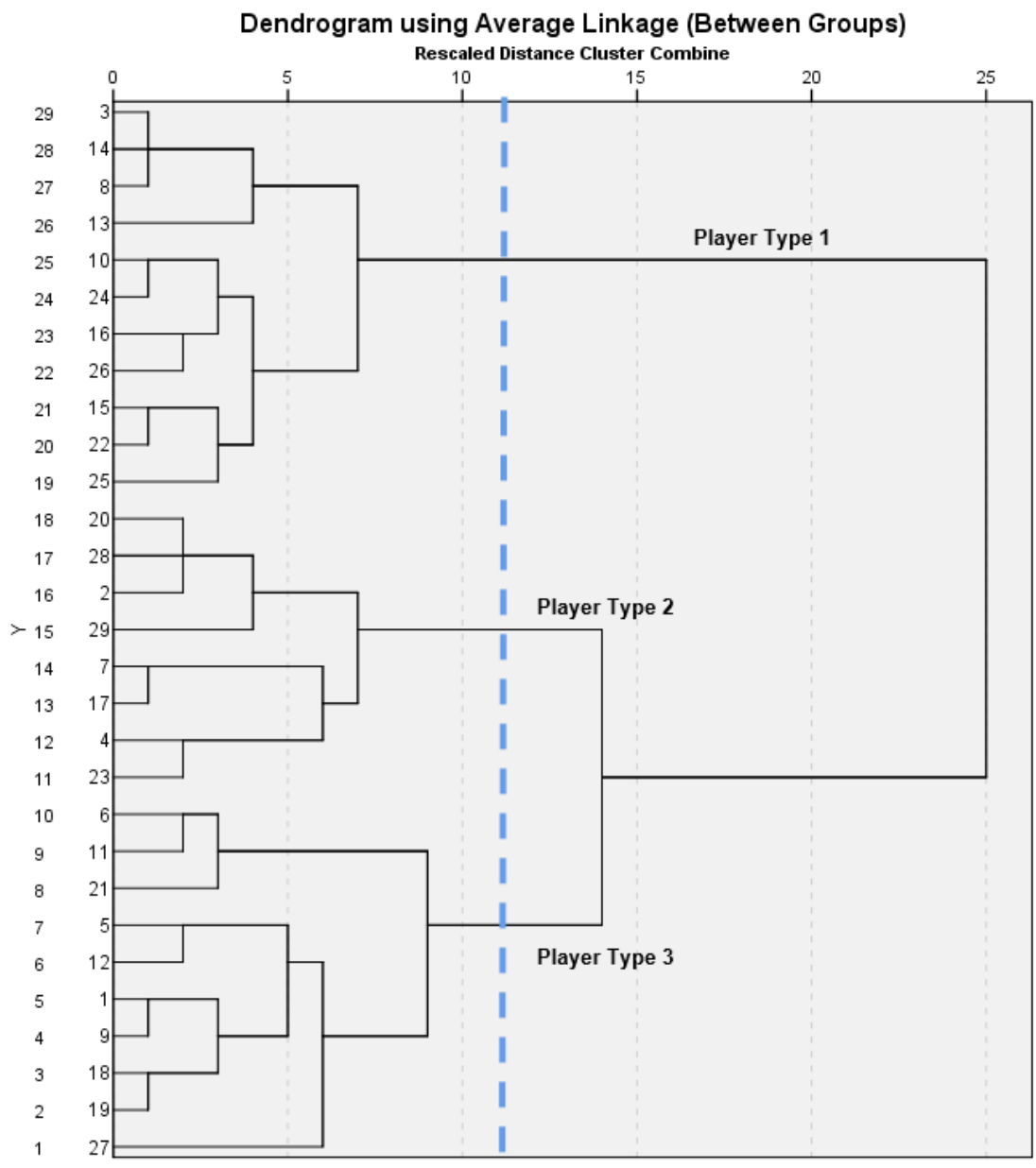
The answers to the questions are based on the Likert Scale 5. The specific questions are as follows :

1.how long is your total time about on playing game ?				
1.0-10hours	2.10-100hours	3.100-500hours	4. 500-1000hours	5. 1000hours 以上
2.The number of games you played is about ? (Only for any games with interactive maps)				
1.0-5	2.6-10	3.11-20	4.20-50	5. more than 50

3.How often do you play on average ?				
1. more than half year	2. 30 days-half year	3. 8-30 days	4. 1-7 days	5. less than 24hours
4.How fast do you think your average behavioral response speed in the game is ?				
1. 慢速	2. 低速	3. 中速	4. 快速	5. 急速
5.When you decide to take an action, how much you think your action range will be ?				
1. 微小	2. 小幅	3. 中幅	4. 大幅	5. 很大幅
6.When you roam the game map, how large the map range you can remember is ?				
1. 微小	2. 小幅	3. 中幅	4. 大幅	5. 很大幅
7.If you are in any game map, when you encounter obstacles in front of you, which action you will choose by default?				
1. 前进	2. 后退	3. 右转	4. 左转	5. 跳跃
8.How fast do you think the expected reaction rate will be when you choose this action ?				
1. 慢速	2. 低速	3. 中速	4. 快速	5. 急速
9.What action range do you think the expected reaction will be when you choose this action ?				
1. 微小	2. 小幅	3. 中幅	4. 大幅	5. 很大幅
10.If you are in any game map, when you encounter unilateral L-shaped obstacles in front of you, which action you will choose by default?				
1. 前进	2. 后退	3. 右转	4. 左转	5. 跳跃
11.How fast do you think the expected reaction rate will be when you choose this action ?				
1. 慢速	2. 低速	3. 中速	4. 快速	5. 急速
12.What action range do you think the expected reaction will be when you choose this action ?				
1. 微小	2. 小幅	3. 中幅	4. 大幅	5. 很大幅
13.If you are in any game map, when you encounter U - shaped obstacles on both sides ahead of you, which action you will choose by default?				
1. 前进	2. 后退	3. 右转	4. 左转	5. 跳跃
14.How fast do you think the expected reaction rate will be when you choose this action ?				
1. 慢速	2. 低速	3. 中速	4. 快速	5. 急速
15.What action range do you think the expected reaction will be when you choose this action ?				
1. 微小	2. 小幅	3. 中幅	4. 大幅	5. 很大幅
16.If you are in any game map, when you encounter upward ladder or stairs in front of you, which action you will choose by default? ?				
1. 前进	2. 后退	3. 右转	4. 左转	5. 跳跃
17.How fast do you think the expected reaction rate will be when you choose this action ?				
1. 慢速	2. 低速	3. 中速	4. 快速	5. 急速
18.What action range do you think the expected reaction will be when you choose this action ?				
1. 微小	2. 小幅	3. 中幅	4. 大幅	5. 很大幅
19.If you are in any game map, when you encounter downward ladder or stairs in front of you, which action you will choose by default? ?				
1. 前进	2. 后退	3. 右转	4. 左转	5. 跳跃
20.How fast do you think the expected reaction rate will be when you choose this action ?				
1. 慢速	2. 低速	3. 中速	4. 快速	5. 急速
21.What action range do you think the expected reaction will be when you choose this action ?				
1. 微小	2. 小幅	3. 中幅	4. 大幅	5. 很大幅
22.If you are in any game map, when you encounter open ground in front of you, which action you will choose by default?				

1.前进	2.后退	3.右转	4.左转	5.跳跃
23.How fast do you think the expected reaction rate will be when you choose this action ?				
1.慢速	2.低速	3.中速	4.快速	5.急速
24.What action range do you think the expected reaction will be when you choose this action ?				
1.微小	2.小幅	3.中幅	4.大幅	5.很大幅
25.If you are in any game map, when you encounter a three-way road, which action you will choose by default?				
1. Random	2. backward	3. Turn right	4. Turn left	5.Take the middle
26.How fast do you think the expected reaction rate will be when you choose this action ?				
1.慢速	2.低速	3.中速	4.快速	5.急速
27.What action range do you think the expected reaction will be when you choose this action ?				
1.微小	2.小幅	3.中幅	4.大幅	5.很大幅





Participants for the survey were recruited at the Beijing University of Posts and Telecommunications in 2021.11; with only one 's age below 20 and varied player experience . About 3/4 of the participants were male, 1/4 female. 34 players participated in the session, and 29 validate answers returned with anomaly checking. Processed by SPSS, the total distribution of the answers show significant 3 clustering (figure 7. The all testers ' behaviors in game maps are classified by hierarchical clustering method according to their game experience. Even in one map state, the behavior choice has a significant relationship with the player type (Figure 5,6). On the one hand, it verifies differentiation theory of player behavior in the references, and can also directly aid to establish three different PSMCTs. The state-action probability of each node on the tree are derived from the sampling probability of a certain type of player. The map memory rate comes from the arithmetic average of such kind players directly (Problem 7).

The test maps was selected from the Beijing University of Posts and Telecommunications in Minecraft into three game scenes with obvious appearance

differences but the same total number of map units. To avoid the birth point effect, all player-like agents are set to appear in the center of the scenes and the RL traversal time are set to a identical 10 minutes.

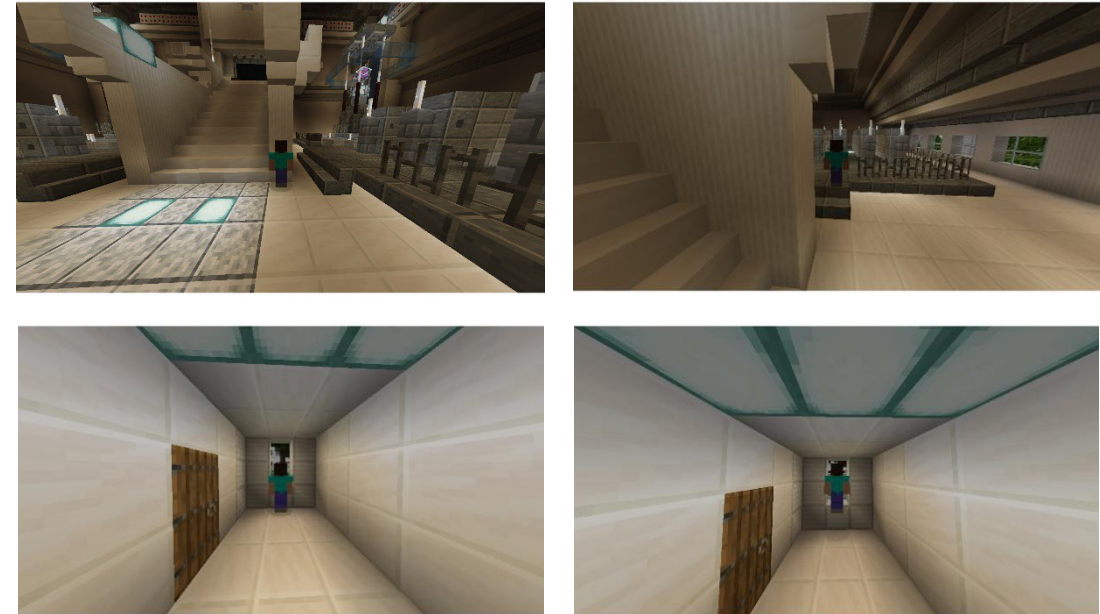
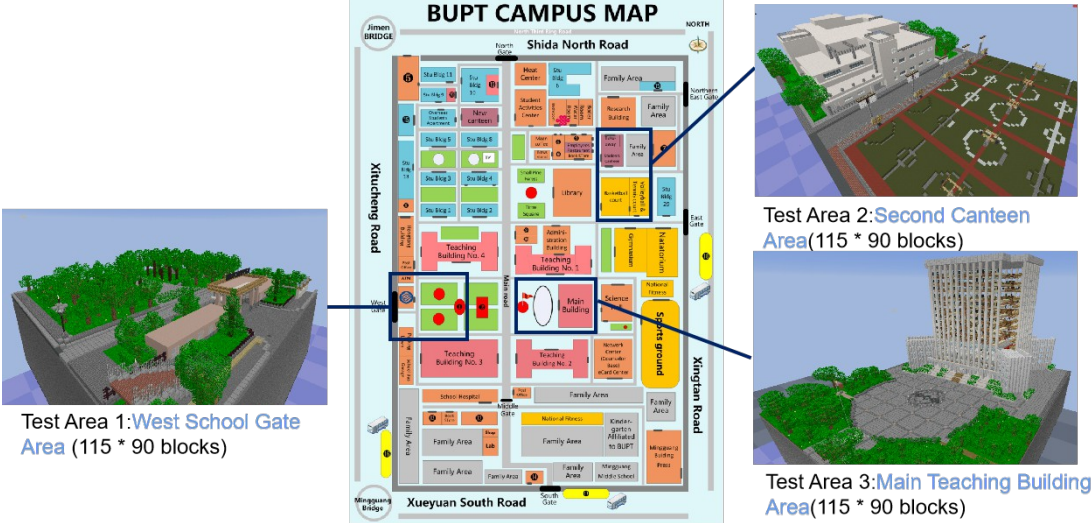


Figure 10: The two images in the first line show how the agent makes the z-axis unchanged action on the flat ground, and the last two show the process of the agent's jumping behavior under the unwalkable situation

Player-like experience of game map	Test Area 1:West School Gate Area	Test Area 2:Second Canteen Area	Test Area 3:Main Teaching Building Area
Agent 1	3250	5667.2	2649
Agent 2	1836	1652	2520
Agent 3	1565	552	1166

In the cross agent-map tests, the final Player-like experience of game map are generally different for each area and each agent type.

From the player-like value table of game map obtained in the experiment, it can be seen that the value of all map areas for Agent 1 (representing players with rich game experience) is relatively high,and the main teaching building area (appearance with the highest spatial complexity) has the highest value comparatively. For Agent 2 (representing players with general game experience), the highest value of the spatial areas is Second Canteen

area, which has both small buildings and flats. The most valuable space map area for Agent 3 (representing novice gamers) is West School Gate area, which is totally flat.

[6] DISCUSSION

Aiming at endless PCG game maps testing for infinite playable value evaluation, this study presents a modified reinforcement learning model to utilize player-like agents to replace human players in map testing, which will greatly reduce the testing workload of human-computer interactions and time or financial cost. the contributions of this study include:

(1) This study proposes a feasible definition of agent behavior for map testing from the perspective of player behavior modeling.

player behavior modeling based on game data and questionnaire survey has been studied in different specific games, but different game types make the models highly complicated. In fact, the agent behavior which is purely used to test the map does not need much complexity. Based on the principles of game space interaction design, this study proposes a special pure behavior tree structure for game map testing. It provides a unified player behavior model for the test of various game maps.

(2) From the perspective of player experience, this study proposes a map value definition model.

Whether agents can obtain experience value in the game, and the convenience and magnitude of obtaining experience value indicate the design quality of game map. Previous studies have coupled specific game types, making it difficult to directly evaluate the design quality of game map without the experience value of other interaction elements. In this study, starting from the commonness of interactive experience of game maps, the overall design quality of spatial maps is decomposed into the cumulative quality of each grid, and the quality of each grid is decomposed into direct exploration and spatial memory according to the theory of game psychology. It provides a unified player experience evaluation model for the test of various game maps.

(3) Based on the model-based reinforcement learning method, this study established a set of reinforcement learning models dedicated to map testing. The reinforcement learning model in this paper has following difference from previous studies.

- 1) The behavior of the agent itself in this study is not variable in the learning process, while the experience value of game map is variable in the iterative learning process. The purpose of reinforcement learning is to automatically enhance the accuracy and comprehensiveness of the experience value evaluation results of map.
- 2) The player experience value of the map is obviously player-oriented. If the agent's behavior model is on behalf of different type of players, the experience value of the identical map is different.
- 3) Effective map evaluation and companion requires limitation on the count of agent actions. Maximizing experiential value is not the goal of training reinforcement learning model in this study. For agents with fixed player-like behavior patterns, unlimited count of actions will improve the experiential value of maps definitely, but this will bias the purpose of map testing. Effective map RL testing must take place in limited time or the count of agent actions in other words.

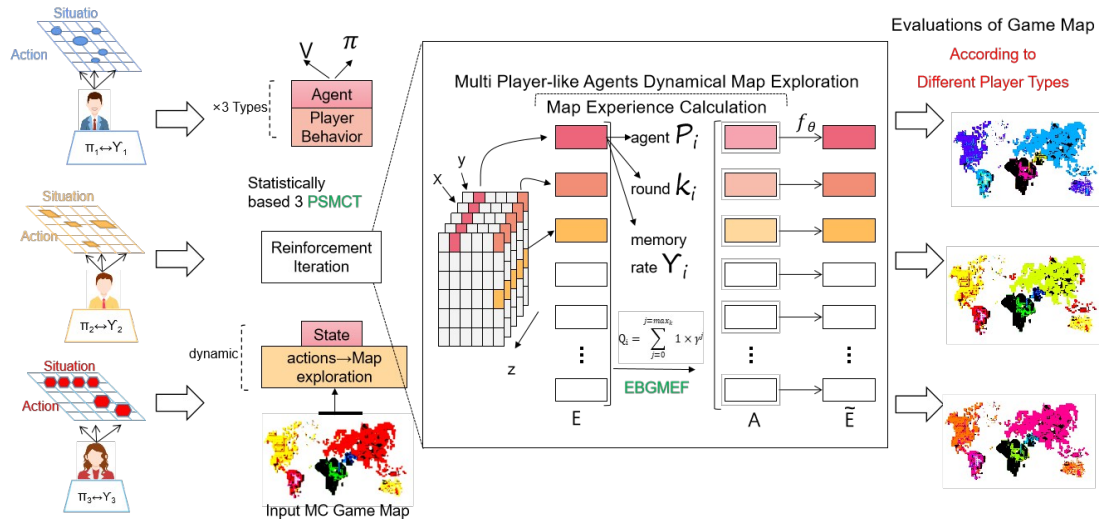


Figure 4: Schematic diagram of player-like agent Reinforcement Learning Method

Through the reinforcement learning model proposed in this study, we can select different types of players to test the map automatically (3 types as shown above). In the experiment of this study, evaluations of identical game map are different according to different types of players, and the total experience value of identical player-like agent to different map is different too, which can effectively evaluate and compare the interactive values of map designs from the perspective of target players of some types. Moreover, the difference between the types of players is more helpful for PCG designers to improve the existed game map, or generate new maps according to the target players of the game map.

There are still two deficiencies in this study :

Firstly, the proposed reinforcement learning model does not couple PCG in iteration for automatic game maps design. In this study, the evaluation of map is independent of the player-like agent 's testing behavior without any spatial structure alteration of the map itself. In the future, PCG game map design can be automatically and intelligently iteratively updated according to RL evaluation to maximum the value of player-like experience, which is also farther this study to the goal of artificial intelligence game design.

Secondly, the experiment in this study is only carried out in Minecraft, in which the calculation of PSMCT and EBGMEF is quite simple compared with other AAA games. Minecraft map is a three-dimensional volume based on the classical octree, and the modeling behaviors of agents are simple too, and the overall calculation workload is much smaller than that of complex 3D maps. But the current mainstream games, especially AAA masterpieces have adopted high precision 3D maps, the player's state-action mode is more complex than Minecraft. If migrating and promoting this study into other types of games , more researches needs to be carried out in the definition of state-action strategy, the map experience calculation and the test computing optimization. In particular, it is also necessary to study the fast extraction method of player's state-action model through game log data [28] [96] to replace the current independent and inefficient player questionnaire.

[7] CONCLUSIONS

From the review of the current literature, game map testing is difficult in matching PCG development with an automatic pattern. Objective and rapid automatic testing can only reflect some superficial indicators of game maps, and cannot evaluate the advantages and disadvantages of game maps from players' perspectives. Game map testing still requires a lot of manual participation, raising the cost of the game industry. While much of the previous literature has focussed on the application of reinforcement learning in games, especially the solutions for AI agents to play various games, there have been few studies which can help game testing.

In general, this study provides new idea and computational frameworks for automated tests of game maps. The contribution of this study is to presents a modified reinforcement learning model combining objective test and subjective test, ensuring the effectiveness of game map test results, including the proposed agent behavior

tree model(PSMCT) and player experience evaluation function for map test(EBGMEF). In the Minecraft experiment, through player surveys, three types agents acting in three test scenes automatically evaluated and scored game maps with distinct player-like perspectives. The experimental results are more subjective than former automatic script map test methods, and more extensive map testing capabilities than some game-specific AI models. Moreover, there is scope for further researches which mix player-like AI test method and PCG method to realize iterative automatic game design, enabling both to co-evolve.

References:

- [1]. R., M.S., et al., A declarative approach to procedural modeling of virtual worlds. *Computers & Graphics*, 2011. 35(2).
- [2]. GSP, M., 1The definition and rendering of terrain maps, in *SIGGRAPH '86: proceedings of the 13th annual conference on computer graphics and interactive techniques*. 1986, ACM: New York, NY, USA.
- [3]. Musgrave, F.K., 2Methods for realistic landscape imaging. 1993, Yale University.
- [4]. Kelley, A.D., M.C. Malin and G.M. Nielson, 3Terrain simulation using a model of stream erosion, in *SIGGRAPH '88: proceedings of the 15th annual conference on computer graphics and interactive techniques*. 1988, ACM: New York, NY, USA.
- [5]. Prusinkiewicz, P. and M. Hammel, 4A fractal model of mountains with rivers. *Proceeding of graphics interface '93*, 1993.
- [6]. Teoh, S.T., 5River and coastal action in automatic terrain generation, in *CGVR 2008: proceedings of the 2008 international conference on CG & VR*. 2008, CSREA Press: Las Vegas, Nevada, USA.
- [7]. Prusinkiewicz, P. and A. Lindenmayer, 6The algorithmic beauty of plants. 1990, New York, NY, USA: Springer-Verlag.
- [8]. Deussen, O., et al., 7Realistic modeling and rendering of plant ecosystems, in *SIGGRAPH '98: proceedings of the 25th annual conference on computer graphics and interactive techniques*. 1998, ACM: New York, NY, USA.
- [9]. Sun, J., et al., 8Template-based generation of road networks for virtual city modeling, in *VRST '02: proceedings of the ACM symposium on virtual reality software and technology*. 2002, ACM: New York, NY, USA.
- [10]. Parish, Y. and P. Muller, 9Procedural modeling of cities, in *SIGGRAPH '01: proceedings of the 28th annual conference on computer graphics and interactive techniques*. 2001, ACM: New York, NY, USA.
- [11]. Wonka, P., et al., 10Instant architecture, in *SIGGRAPH '03: proceedings of the 30th annual conference on computer graphics and interactive techniques*. 2003, ACM: New York, NY, USA.
- [12]. Muller, P., et al., 11Procedural modeling of buildings, in *SIGGRAPH '06: proceedings of the 33rd annual conference on computer graphics and interactive techniques*. 2006, ACM: New York, NY, USA.
- [13]. Finkenzeller, D., 12Detailed building facades. *IEEE Computer Graphics and Applications* 2008, 2008.
- [14]. Dumim, Y. and K. Kyung-Joong, 3D Game Model and Texture Generation Using Interactive Genetic Algorithm. *Computers in Entertainment (CIE)*, 2016. 14(1).
- [15]. Smelik, R.M., et al., A Survey on Procedural Modelling for Virtual Worlds. *Computer Graphics Forum*, 2014. 33(6): p. 31-50.
- [16]. Tessler, C., et al. A Deep Hierarchical Approach to Lifelong Learning in Minecraft. in *AAAI Publications, Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)*. 2016.
- [17]. Cook, M., S. Colton and J. Gow, Automating Game Design In Three Dimensions. 2014.
- [18]. Bruneton, E. and F. Neyret, 33Real-time rendering and editing of vector-based terrains, in *Computer graphics forum: eurographics 2008 proceedings*. 2008: Crete, Greece.
- [19]. Vanegas, C.A., et al., 30 Interactive design of urban spaces using geometrical and behavioral modeling, in *ACM TOG: proceedings of ACM SIGGRAPH Asia*. 2009, ACM.

- [20]. J., T., et al., Search-Based Procedural Content Generation: A Taxonomy and Survey. *IEEE Transactions on Computational Intelligence and AI in Games*, 2011. 3(3): p. 172-186.
- [21]. Huang, H., Intelligent Pathfinding Algorithm in Web Games. 2020: Cyber Security Intelligence and Analytics.
- [22]. Machado, P. and A. Cardoso. Computing aesthetics. in *Lecture Notes in Artificial Intelligence*. 1998.
- [23]. Sims and Karl, Artificial evolution for computer graphics. *ACM*, 1991: p. 319-328.
- [24]. Secretan, J., et al. Picbreeder: evolving pictures collaboratively online. in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2008.
- [25]. Khaled, R., M.J. Nelson and P. Barr. Design metaphors for procedural content generation in games. in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2013.
- [26]. Kazmi, S. and I.J. Palmer, Action Recognition for Support of Adaptive Gameplay: A Case Study of a First Person Shooter. *International Journal of Computer Games Technology*, 2010.
- [27]. Sims, K., Interactive evolution of equations for procedural models. *The Visual Computer*, 2005. 9: p. 466-476.
- [28]. Vitek, A. Cross-Game Modeling of Player's Behaviour in Free-To-Play Games. in *Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization*. 2020. Genoa, Italy: Association for Computing Machinery.
- [29]. Roohi, S., et al., Predicting Game Difficulty and Engagement Using AI Players. *Proc. ACM Hum.-Comput. Interact.*, 2021. 5(CHI PLAY): p. Article 231.
- [30]. Zhu, J. and S. Ontañón. Player-Centered AI for Automatic Game Personalization: Open Problems. in *International Conference on the Foundations of Digital Games*. 2020. Bugibba, Malta: Association for Computing Machinery.
- [31]. Ariyurek, S., A. Betin-Can and E. Surer, Automated Video Game Testing Using Synthetic and Human-Like Agents. 2019.
- [32]. Fernando, D., et al. AI-based playtesting of contemporary board games. in *International Conference*. 2017.
- [33]. Isaksen, A., G. Dan and A. Nealen, Exploring Game Space Using Survival Analysis. 2015.
- [34]. Harmer, J., et al. Imitation Learning with Concurrent Actions in 3D Games. in *2018 IEEE Conference on Computational Intelligence and Games (CIG)*. 2018.
- [35]. Lample, G. and D.S. Chaplot. Playing FPS Games with Deep Reinforcement Learning. in *31st AAAI Conference on Artificial Intelligence (AAAI-17)*, San Francisco, USA. 2016.
- [36]. Cowling, P.I., Combining Gameplay Data with Monte Carlo Tree Search to Emulate Human Play. 2016.
- [37]. Zook, A., B. Harrison and M.O. Riedl, Monte-Carlo Tree Search for Simulation-based Strategy Analysis. 2019.
- [38]. Nelson, M.J. Investigating Vanilla MCTS Scaling on the GVG-AI Game Corpus. in *2016 IEEE Conference on Computational Intelligence and Games (CIG)*. 2017.
- [39]. Khalifa, A., et al., Modifying MCTS for Human-Like General Video Game Playing. *AAAI Press*, 2016.
- [40]. Holmgård, C., et al., Automated Playtesting With Procedural Personas Through MCTS With Evolved Heuristics. *IEEE Transactions on Games*, 2019. 11(4): p. 352-362.
- [41]. Bergdahl, J., et al., Augmenting Automated Game Testing with Deep Reinforcement Learning. 2021.
- [42]. Bakkes, S., P. Spronck and G.V. Lankveld, Player behavioural modelling for video games. *Entertainment Computing*, 2012. 3(3): p. 71-79.
- [43]. Drachen, A. and M. Schubert, Spatial game analytics and visualization. 2013. p. 1-8.
- [44]. Frago, L. and K.G. Stanley, StABLE: Analyzing Player Movement Similarity Using Text Mining.
- [45]. Laviola, J.J. and R.L. Marks, An introduction to 3D spatial interaction with video game motion controllers. *ACM*, 2010: p. 1-78.
- [46]. Jr, L.V. and D.F. Keefe. Course: 3D Spatial Interaction: Applications for Art, Design, and Science. in *ACM SIGGRAPH 2011 ACM SIGGRAPH 2011*. 2011.
- [47]. Mueller, S., et al. HeapCraft: interactive data exploration and visualization tools for understanding and influencing player behavior in Minecraft. in *the 8th ACM SIGGRAPH Conference*. 2015.
- [48]. Thawonmas, R., et al. Clustering of Online Game Users Based on Their Trails Using Self-organizing Map. in *Entertainment Computing - ICEC 2006, 5th International Conference, Cambridge, UK, September 20-22, 2006, Proceedings*. 2006.
- [49]. Melhart, D., A. Liapis and A.G.N. Yannakakis., Towards General Models of Player Experience: A Study Within Genres., in *IEEE Conference on Games*. Retrieved from

<https://again.institutedigitalgames.com/>. 2021.

- [50]. Gratch, J.M. and S.C. Marsella, Evaluating a Computational Model of Emotion. *Autonomous Agents and Multi-Agent Systems*, 2005.
- [51]. Conati, C., Probabilistic assessment of user's emotions in educational games. *Applied Artificial Intelligence*, 2002. 16(7): p. 555-575.
- [52]. Drachen, A., et al., Correlation between heart rate, electrodermal activity and player experience in first-person shooter games. *Dragon Consulting; Department of Computer Science University of Saskatchewan; Center for Computer Games Research IT University Copenhagen; Center for Computer Games Research IT University of Copenhagen*, 2011.
- [53]. Pagulayan, R.J., et al., *User-centered design in games*. L. Erlbaum Associates Inc., 2002.
- [54]. Yannakakis, G.N. Preference learning for affective modeling. in *2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops*. 2009.
- [55]. Frijda, N.H., The Emotions. *Studies in Emotion & Social Interaction*, 1986. 1(5): p. 583-584.
- [56]. Isbister, K., *Game Usability*. 2008.
- [57]. Ortony, A., G.L. Clore and A. Collins, The Cognitive Structure of Emotions. *Contemporary Sociology*, 1988. 18(6): p. 2147-2153.
- [58]. Skinner, B.F., *The behavior of organisms: An experimental analysis*. appleton century new york smith a, 1938.
- [59]. Scherer and R. Klaus, Studying the emotion-antecedent appraisal process: An expert system approach. *Cognition & Emotion*, 1993. 7(3-4): p. 325-355.
- [60]. Malone, T.W. What makes things fun to learn? heuristics for designing instructional computer games. in *Acm Sigsmall Symposium & the First Sigpc Symposium on Small Systems*. 1980.
- [61]. Koster, R., *A Theory of Fun for Game Design*. Paraglyph Press, 2004.
- [62]. Csikszentmihalyi, M., Flow: The Psychology of Optimal Experience. *Design Issues*, 1991. 8(1).
- [63]. Feldman, L.A., Valence Focus and Arousal Focus: Individual Differences in the Structure of Affective Experience. *Journal of Personality and Social Psychology*, 1995. 69(1): p. 153-166.
- [64]. Russell, J.A., Core Affect and the Psychological Construction of Emotion. *Journal of Behavioral Finance*, 2003.
- [65]. Asteriadis, S., et al., A natural head pose and eye gaze dataset. *ACM*, 2009: p. 1-4.
- [66]. Shaker, N., G.N. Yannakakis and J. Togelius. Towards Automatic Personalized Content Generation for Platform Games. in *Proceedings of the Sixth AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, AIIDE 2010, October 11-13, 2010, Stanford, California, USA*. 2010.
- [67]. Ferro, L.S., The Game Element and Mechanic (GEM) Framework: a structural approach for implementing game elements and mechanics into game experiences. *Entertainment Computing*, 2020.
- [68]. Frago, S. Interface design strategies and disruptions of gameplay: notes from a qualitative study with first-person gamers. in *International Conference on Human-Computer Interaction*. 2014.
- [69]. Touns, Z., et al., Making Maps Available for Play: Analyzing the Design of Game Cartography Interfaces. *ACM transactions on computer-human interaction*, 2019. 26(5): p. 1-43.
- [70]. Matsumoto, Y. and R. Thawonmas. MMOG Player Classification Using Hidden Markov Models. 2004. Berlin, Heidelberg: Springer Berlin Heidelberg.
- [71]. Stahlke, S.N. and P. Mirza-Babaei, Usertesting Without the User: Opportunities and Challenges of an AI-Driven Approach in Games User Research. *Comput. Entertain.*, 2018. 16(2).
- [72]. Ashlock, D. and C. Salge, Automatic Generation of Level Maps with the Do What's Possible Representation. 2019.
- [73]. Mnih, V., et al., Playing Atari with Deep Reinforcement Learning. *Computer Science*, 2013.
- [74]. Berner, C., et al., Dota 2 with Large Scale Deep Reinforcement Learning. 2019.
- [75]. Doll, B.B., D.A. Simon and N.D. Daw, The ubiquity of model-based reinforcement learning. *Current opinion in neurobiology*, 2012. 22(6): p. 1075-1081.
- [76]. Kaiser, L., et al., Model-Based Reinforcement Learning for Atari. *ArXiv*, 2020. abs/1903.00374.
- [77]. Hafner, D., et al., Learning Latent Dynamics for Planning from Pixels. 2018.
- [78]. Piergiovanni, A.J., A. Wu and M.S. Ryoo, Learning Real-World Robot Policies by Dreaming. 2018.
- [79]. Rybkin, O., et al., Unsupervised Learning of Sensorimotor Affordances by Stochastic Future Prediction. 2018.
- [80]. Ebert, F., et al., Self-Supervised Visual Planning with Temporal Skip Connections. 2017.
- [81]. Finn, C., et al. Deep spatial autoencoders for visuomotor learning. in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. 2016.
- [82]. Watter, M., et al., Embed to Control: A Locally Linear Latent Dynamics Model for Control from Raw Images. *Advances in neural information processing systems*, 2015.

- [83]. Ha, D. and J. Schmidhuber, Recurrent World Models Facilitate Policy Evolution. 2018.
- [84]. Jeong, K. and J. Choi, Deep Recurrent Neural Network. Communications of the Korean Institute of Information Scientists and Engineers, 2015. 33.
- [85]. Schmidhuber, J., Formal Theory of Fun and Creativity. DBLP, 2010.
- [86]. Buesing, L., et al., Woulda, Coulda, Shoulda: Counterfactually-Guided Policy Search. 2018.
- [87]. Nebel, S., S. Schneider and G.D. Rey, Mining Learning and Crafting Scientific Experiments: A Literature Review on the Use of Minecraft in Education and Research. Journal of Educational Technology & Society, 2016. 19(2): p. 355-366.
- [88]. Salge, C., et al., Generative Design in Minecraft: Chronicle Challenge. 2019.
- [89]. Sharma, M., et al., Player modeling evaluation for interactive fiction. 2009.
- [90]. Moll, P., et al., How Players Play Games: Observing the Influences of Game Mechanics. 2019.
- [91]. Stefan, S., et al., Learning human-like Movement Behavior for Computer Games. 2004, MIT Press. p. 315-323.
- [92]. Bauckhage, C., A. Drachen and R. Sifa, Clustering Game Behavior Data. IEEE Transactions on Computational Intelligence and AI in Games, 2015. 7(3): p. 266-278.
- [93]. Brühlmann, F. and E. Mekler, Surveys in Games User Research. 2018.
- [94]. Sifa, R., A. Drachen and C. Bauckhage. Large-Scale Cross-Game Player Behavior Analysis on Steam. in AIIDE. 2015.
- [95]. Lopes, R., T. Tutenel and R. Bidarra. Using gameplay semantics to procedurally generate player-matching game worlds. in Proceedings of the The third workshop on Procedural Content Generation in Games. 2012. Raleigh, NC, USA: Association for Computing Machinery.